

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings of claims in the application:

LISTING OF CLAIMS

1. (Currently Amended) A small footprint device comprising:
- at least one processing element configured to execute groups of one or more program modules in separate contexts, said one or more program modules comprising zero or more sets of executable instructions and zero or more sets of data definitions, said zero or more sets of executable instructions and said zero or more data definitions grouped as object definitions, each context comprising a protected object instance space such that at least one of said object definitions is instantiated in association with a particular context;
- a memory comprising instances of objects;
- a context barrier for separating and isolating ~~program modules from one another using said memory and said processing element, said program modules configured to operate on said small footprint device~~ said contexts, said context barrier configured for controlling execution of at least one instruction of one of said zero or more sets of instructions comprised by a program module based at least in part on whether said at least one instruction is executed for an object instance associated with a first one of said one or more separate contexts and whether said at least one instruction is requesting access to an instance of an object definition associated with a second one of said one or more separate

contexts, said context barrier further configured to prevent said access if said access is unauthorized and enable said access if said access is authorized; and
one context having access to all program modules without context barrier constraints.

2-24. (Cancelled)

25. (Previously Presented) The small footprint device of claim 1 in which said context is used for access to at least one program module across a context barrier.

26. (Previously Presented) The small footprint device of claim 1 in which said context allocates separate name spaces for each program module.

27. (Previously Presented) The small footprint device of claim 26 in which said context can access at least one other program module even though it is located in a different name space.

28. (Previously Presented) The small footprint device of claim 1 in which said context allocates separate memory spaces for each program module.

29. (Previously Presented) The small footprint device of claim 28 in which said context can access at least one program module even though it is located in a different memory space.

30. (Previously Presented) The small footprint device of claim 1 in which said context barrier enforces security checks on at least one of a principal, an object and an action.

31. (Previously Presented) The small footprint device of claim 30 in which at least one security check is based on partial name agreement between a principal and an object.
32. (Previously Presented) The small footprint device of claim 31 in which said context can access at least one other context without said at least one security check.
33. (Previously Presented) The small footprint device of claim 30 in which at least one security check is based on memory space agreement between a principal and an object.
34. (Previously Presented) The small footprint device of claim 33 in which said context can access at least one other context without said at least one security check.
35. (Currently Amended) A method of operating a small footprint device that includes a processing machine, wherein program modules are executed on the processing machine, the method, comprising:
separating program modules contexts using a context barrier, said program modules context barrier configured to operate on said small footprint device for controlling execution of at least one instruction of one of said zero or more sets of instructions comprised by a program module based at least in part on whether said at least one instruction is executed for an object instance associated with a first one of said one or more separate contexts and whether said at least one instruction is requesting access to an instance of

an object definition associated with a second one of said one or more separate contexts,
said separating further comprising:
preventing said access if said access is unauthorized; and
enabling said access if said access is authorized; and
executing groups of one or more program modules in separate contexts, said one or more
program modules comprising zero or more sets of executable instructions and zero or
more sets of data definitions, said zero or more sets of executable instructions and said
zero or more data definitions grouped as object definitions, each context comprising a
protected object instance space such that at least one of said object definitions is
instantiated in association with a particular context; and
permitting one context in said small footprint device access to at least one other context in
said small footprint device without context barrier constraints.

36. (Previously Presented) The method of claim 35 in which the context barrier will not permit
a principal to perform an action on an object unless both principal and object are part of the
same context or the principal is part of said one context.

37. (Currently Amended) A method of permitting access to information on a small footprint
device from a first program module to a second program module separated by a context
barrier, the small footprint device comprising:

at least one processing element configured to execute groups of one or more program
modules in separate contexts, said one or more program modules comprising zero or
more sets of executable instructions and zero or more sets of data definitions, said zero

or more sets of executable instructions and said zero or more data definitions grouped as
object definitions, each context comprising a protected object instance space such that at
least one of said object definitions is instantiated in association with a particular context
a memory comprising instances of objects; and
a context barrier for separating and isolating said contexts, said context barrier configured
for controlling execution of at least one instruction of one of said zero or more sets of
instructions comprised by a program module based at least in part on whether said at
least one instruction is executed for an object instance associated with a first one of said
one or more separate contexts and whether said at least one instruction is requesting
access to an instance of an object definition associated with a second one of said one or
more separate contexts, said context barrier further configured to prevent said access if
said access is unauthorized and enable said access if said access is authorized, the
method comprising:

creating a context in said small footprint device, said context having access to all program
modules without context barrier constraints, ~~said program modules configured to operate~~
~~on said small footprint device.~~

38. (Previously Presented) The method of claim 37 in which said context is a supercontext.

39. (Currently Amended) A method of communicating across a context barrier separating
program modules on a small footprint device, the small footprint device comprising:
at least one processing element configured to execute groups of one or more program
modules in separate contexts, said one or more program modules comprising zero or

more sets of executable instructions and zero or more sets of data definitions, said zero or more sets of executable instructions and said zero or more data definitions grouped as object definitions, each context comprising a protected object instance space such that at least one of said object definitions is instantiated in association with a particular context a memory comprising instances of objects; and
a context barrier for separating and isolating said contexts, said context barrier configured for controlling execution of at least one instruction of one of said zero or more sets of instructions comprised by a program module based at least in part on whether said at least one instruction is executed for an object instance associated with a first one of said one or more separate contexts and whether said at least one instruction is requesting access to an instance of an object definition associated with a second one of said one or more separate contexts, said context barrier further configured to prevent said access if said access is unauthorized and enable said access if said access is authorized, the method comprising:

creating a context in said small footprint device, said context having access to all program modules without context barrier constraints, ~~said program modules configured to operate on said small footprint device;~~ and
permitting said context to access information of another program module across said context barrier.

40. (Currently Amended) A method of communicating across a context barrier separating program modules on a small footprint device, the small footprint device comprising:

at least one processing element configured to execute groups of one or more program modules in separate contexts, said one or more program modules comprising zero or more sets of executable instructions and zero or more sets of data definitions, said zero or more sets of executable instructions and said zero or more data definitions grouped as object definitions, each context comprising a protected object instance space such that at least one of said object definitions is instantiated in association with a particular context
a memory comprising instances of objects; and
a context barrier for separating and isolating said contexts, said context barrier configured for controlling execution of at least one instruction of one of said zero or more sets of instructions comprised by a program module based at least in part on whether said at least one instruction is executed for an object instance associated with a first one of said one or more separate contexts and whether said at least one instruction is requesting access to an instance of an object definition associated with a second one of said one or more separate contexts, said context barrier further configured to prevent said access if said access is unauthorized and enable said access if said access is authorized, the method comprising:

creating a context in said small footprint device, said context having access to all program modules without context barrier constraints, ~~said program modules configured to operate on said small footprint device;~~ and
permitting at least one program module to access information of another program module across said context barrier using said context.

41. (Currently Amended) A computer program product, comprising:

a memory medium; and

a computer controlling element comprising instructions for implementing a context barrier

on a small footprint device and for giving one context in said small footprint device

access to all program modules without context barrier constraints, ~~said program~~

~~modules configured to operate on said small footprint device.~~ said small footprint

device comprising:

at least one processing element configured to execute groups of one or more program

modules in separate contexts, said one or more program modules comprising zero or

more sets of executable instructions and zero or more sets of data definitions, said zero

or more sets of executable instructions and said zero or more data definitions grouped as

object definitions, each context comprising a protected object instance space such that at

least one of said object definitions is instantiated in association with a particular context;

a memory comprising instances of objects; and

a context barrier for separating and isolating said contexts, said context barrier configured

for controlling execution of at least one instruction of one of said zero or more sets of

instructions comprised by a program module based at least in part on whether said at

least one instruction is executed for an object instance associated with a first one of said

one or more separate contexts and whether said at least one instruction is requesting

access to an instance of an object definition associated with a second one of said one or

more separate contexts, said context barrier further configured to prevent said access if

said access is unauthorized and enable said access if said access is authorized.

42. (Previously Presented) The computer program product of claim 41, in which said medium is a carrier wave.

43. (Currently Amended) A computer program product, comprising:

a memory medium; and

a computer controlling element comprising instructions for separating a plurality of

programs on a small footprint device by running them in respective contexts and for

permitting one context in said resource constrained device to have access to all program

modules without context barrier constraints, ~~said program modules configured to~~

~~operate on said small footprint device. said program modules configured to operate on~~

~~said small footprint device. said small footprint device comprising:~~

at least one processing element configured to execute groups of one or more program

modules in separate contexts, said one or more program modules comprising zero or

more sets of executable instructions and zero or more sets of data definitions, said zero

or more sets of executable instructions and said zero or more data definitions grouped as

object definitions, each context comprising a protected object instance space such that at

least one of said object definitions is instantiated in association with a particular context;

a memory comprising instances of objects; and

a context barrier for separating and isolating said contexts, said context barrier configured

for controlling execution of at least one instruction of one of said zero or more sets of

instructions comprised by a program module based at least in part on whether said at

least one instruction is executed for an object instance associated with a first one of said

one or more separate contexts and whether said at least one instruction is requesting

access to an instance of an object definition associated with a second one of said one or more separate contexts, said context barrier further configured to prevent said access if said access is unauthorized and enable said access if said access is authorized.

44. (Previously Presented) The computer program product of claim 43, in which said medium is a carrier wave.

45. (Currently Amended) A carrier wave carrying instructions over a communications link for implementing a context in said resource constrained device, said context having access to all program modules on a small footprint device without context barrier constraints, ~~said program modules configured to operate on said small footprint device.~~ said small footprint device comprising:

at least one processing element configured to execute groups of one or more program modules in separate contexts, said one or more program modules comprising zero or more sets of executable instructions and zero or more sets of data definitions, said zero or more sets of executable instructions and said zero or more data definitions grouped as object definitions, each context comprising a protected object instance space such that at least one of said object definitions is instantiated in association with a particular context;

a memory comprising instances of objects; and

a context barrier for separating and isolating said contexts, said context barrier configured for controlling execution of at least one instruction of one of said zero or more sets of instructions comprised by a program module based at least in part on whether said at least one instruction is executed for an object instance associated with a first one of said

one or more separate contexts and whether said at least one instruction is requesting access to an instance of an object definition associated with a second one of said one or more separate contexts, said context barrier further configured to prevent said access if said access is unauthorized and enable said access if said access is authorized.

46. (Currently Amended) A carrier wave carrying instructions over a communications link for implementing a context barrier separating a plurality of programs on a small footprint device by running them in respective contexts and for permitting one context in said resource constrained device to access all programs without context barrier constraints, ~~said programs configured to operate on said small footprint device.~~ said small footprint device comprising:

at least one processing element configured to execute groups of one or more program modules in separate contexts, said one or more program modules comprising zero or more sets of executable instructions and zero or more sets of data definitions, said zero or more sets of executable instructions and said zero or more data definitions grouped as object definitions, each context comprising a protected object instance space such that at least one of said object definitions is instantiated in association with a particular context; a memory comprising instances of objects; and a context barrier for separating and isolating said contexts, said context barrier configured for controlling execution of at least one instruction of one of said zero or more sets of instructions comprised by a program module based at least in part on whether said at least one instruction is executed for an object instance associated with a first one of said one or more separate contexts and whether said at least one instruction is requesting

access to an instance of an object definition associated with a second one of said one or more separate contexts, said context barrier further configured to prevent said access if said access is unauthorized and enable said access if said access is authorized.

47. (Currently Amended) A method of transmitting code over a network, comprising transmitting a block of code from a server, said block of code comprising instructions for implementing a context in said small footprint device, said context having access to all program modules for providing access across a context barrier, ~~said program modules configured to operate on said small footprint device.~~ said small footprint device comprising: at least one processing element configured to execute groups of one or more program modules in separate contexts, said one or more program modules comprising zero or more sets of executable instructions and zero or more sets of data definitions, said zero or more sets of executable instructions and said zero or more data definitions grouped as object definitions, each context comprising a protected object instance space such that at least one of said object definitions is instantiated in association with a particular context; a memory comprising instances of objects; and a context barrier for separating and isolating said contexts, said context barrier configured for controlling execution of at least one instruction of one of said zero or more sets of instructions comprised by a program module based at least in part on whether said at least one instruction is executed for an object instance associated with a first one of said one or more separate contexts and whether said at least one instruction is requesting access to an instance of an object definition associated with a second one of said one or

more separate contexts, said context barrier further configured to prevent said access if said access is unauthorized and enable said access if said access is authorized.

48. (Currently Amended) An apparatus for communicating across a context barrier separating program modules on a small footprint device, said small footprint device comprising:
- at least one processing element configured to execute groups of one or more program modules in separate contexts, said one or more program modules comprising zero or more sets of executable instructions and zero or more sets of data definitions, said zero or more sets of executable instructions and said zero or more data definitions grouped as object definitions, each context comprising a protected object instance space such that at least one of said object definitions is instantiated in association with a particular context;
- a memory comprising instances of objects; and
- a context barrier for separating and isolating said contexts, said context barrier configured for controlling execution of at least one instruction of one of said zero or more sets of instructions comprised by a program module based at least in part on whether said at least one instruction is executed for an object instance associated with a first one of said one or more separate contexts and whether said at least one instruction is requesting access to an instance of an object definition associated with a second one of said one or more separate contexts, said context barrier further configured to prevent said access if said access is unauthorized and enable said access if said access is authorized, said apparatus comprising:

means for creating a context in said small footprint device, said context having access to all program modules without context barrier constraints, ~~said program modules configured to operate on said small footprint device~~; and

means for permitting said context to access information of another program module across said context barrier.

49. (Currently Amended) An apparatus for communicating across a context barrier separating program modules on a small footprint device, said small footprint device comprising:
- at least one processing element configured to execute groups of one or more program modules in separate contexts, said one or more program modules comprising zero or more sets of executable instructions and zero or more sets of data definitions, said zero or more sets of executable instructions and said zero or more data definitions grouped as object definitions, each context comprising a protected object instance space such that at least one of said object definitions is instantiated in association with a particular context;
- a memory comprising instances of objects; and
- a context barrier for separating and isolating said contexts, said context barrier configured for controlling execution of at least one instruction of one of said zero or more sets of instructions comprised by a program module based at least in part on whether said at least one instruction is executed for an object instance associated with a first one of said one or more separate contexts and whether said at least one instruction is requesting access to an instance of an object definition associated with a second one of said one or more separate contexts, said context barrier further configured to prevent said access if

said access is unauthorized and enable said access if said access is authorized, said

apparatus comprising:

means for creating a context in said small footprint device, said context having access to all
program modules without context barrier constraints; ~~said program modules configured
to operate on said small footprint device; and~~

means for permitting at least one program module to access information of another program
module across said context barrier using said context.

50. (New) The small footprint device of claim 1 wherein an object instance is associated with a
context by recording the name of said context in a header of said object instance,
information in said header inaccessible to said one or more program modules.

51. (New) The small footprint device of claim 1 wherein
said memory comprises object header data, said object header data comprising information
associated with at least one of said instances of objects; and
said controlling execution is based at least in part on said object header data.

52. (New) The small footprint device of claim 1 wherein
said memory is partitioned into a plurality of memory spaces with instances of objects
allocated for storage in one of said plurality of storage spaces; and
said controlling execution is based at least in part on determining the storage space allocated
to an executing object instance and an accessed object instance.

53. (New) The method of claim 35 wherein an object instance is associated with a context by recording the name of said context in a header of said object instance, information in said header inaccessible to said one or more program modules.
54. (New) The method of claim 35 wherein said controlling execution is based at least in part on object header data comprising information associated with at least one of said instances of objects.
55. (New) The method of claim 35 wherein
a memory of said small footprint device is partitioned into a plurality of memory spaces with instances of objects allocated for storage in one of said plurality of storage spaces; and said controlling execution is based at least in part on determining the storage space allocated to an executing object instance and an accessed object instance.
56. (New) The method of claim 37 wherein an object instance is associated with a context by recording the name of said context in a header of said object instance, information in said header inaccessible to said one or more program modules.
57. (New) The method of claim 37 wherein said controlling execution is based at least in part on object header data comprising information associated with at least one of said instances of objects.
58. (New) The method of claim 37 wherein

a memory of said small footprint device is partitioned into a plurality of memory spaces with instances of objects allocated for storage in one of said plurality of storage spaces; and said controlling execution is based at least in part on determining the storage space allocated to an executing object instance and an accessed object instance.

59. (New) The method of claim 39 wherein an object instance is associated with a context by recording the name of said context in a header of said object instance, information in said header inaccessible to said one or more program modules.

60. (New) The method of claim 39 wherein said controlling execution is based at least in part on object header data comprising information associated with at least one of said instances of objects.

61. (New) The method of claim 39 wherein
a memory of said small footprint device is partitioned into a plurality of memory spaces with instances of objects allocated for storage in one of said plurality of storage spaces; and said controlling execution is based at least in part on determining the storage space allocated to an executing object instance and an accessed object instance.

62. (New) The method of claim 40 wherein an object instance is associated with a context by recording the name of said context in a header of said object instance, information in said header inaccessible to said one or more program modules.

63. (New) The method of claim 40 wherein said controlling execution is based at least in part on object header data comprising information associated with at least one of said instances of objects.

64. (New) The method of claim 40 wherein
a memory of said small footprint device is partitioned into a plurality of memory spaces with instances of objects allocated for storage in one of said plurality of storage spaces; and said controlling execution is based at least in part on determining the storage space allocated to an executing object instance and an accessed object instance.
